Student Information System

Requirement Specifications

March 11, 2025

–

UIS & Student Community

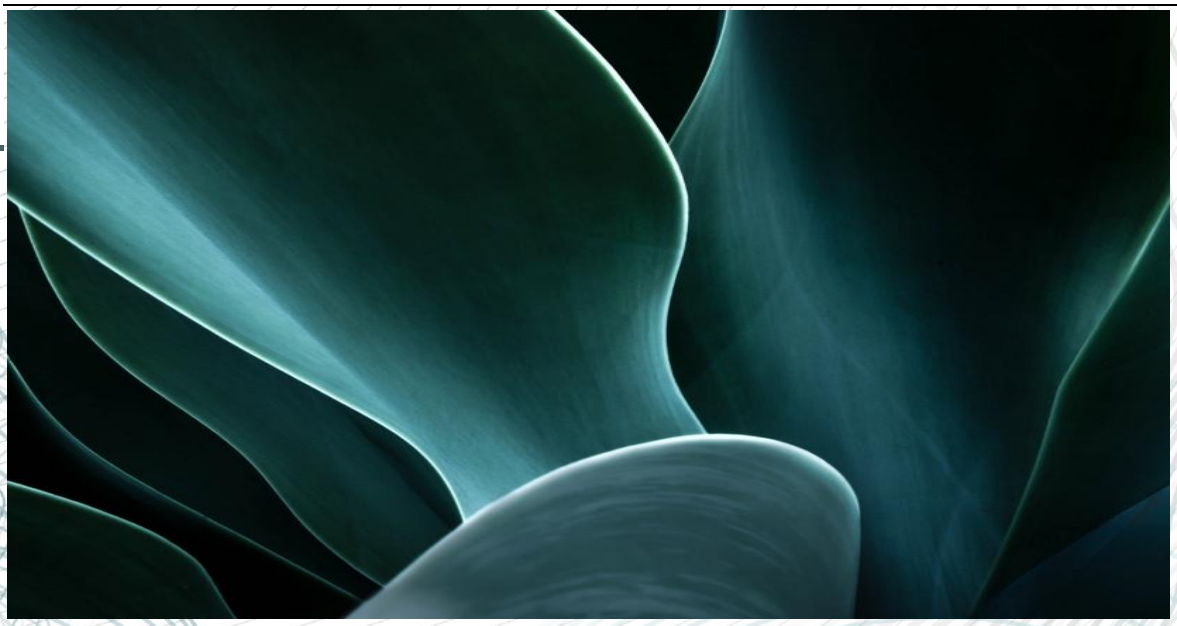
–

Epoka University

Table of content

Project Overview

Our vision is to develop a student-centric system, designed to prioritize their needs and experiences. This system will encompass two integrated software solutions: one dedicated to the internal management of university operations and the other tailored to fostering a vibrant student community. These software applications will be offered as a comprehensive package. Universities that adopt this system will provide their enrolled students with full access to both platforms. However, individuals not affiliated with a participating university will still be able to access the community software independently.



University Information System (UIS)

Product Context

The first software is for the internal management of the university. This system is bought by the universities in the Albania.

The ***stakeholders*** of this system include:

- Students
- Academic Staff
- Finance Office
- Admissions Office
- Career Office
- ICT Coordination
- Registrar's Office

User Characteristics:

- User Authentication and Profile Management ensures secure access and management of user accounts and profiles.
- **Students** can access grades, selected courses, financial details, attendance records, transcripts, and student guidelines, and can enroll in classrooms.
- **Admissions Office** uploads the primary data of matura students upon signing their university entry contracts.
- **Academic staff** manages the syllabus, classroom materials, attendance records, grades, and perform academic reporting.
- **Finance Office** oversees student billing, payment tracking, financial aid management (including scholarships), and generates financial reports.
- **Career Office** facilitates job and internship postings, resume and interview preparation, and alumni networking.
- **Registrar's Office** maintains student records, uploads additional exam grades, processes exam requests, handles transcript requests, schedules courses, and performs degree audits.
- **Dean of Students** publishes student guides, fosters involvement in extracurricular activities, and generates club activity reports.

ThinkLink – Connecting through learning

Product Context

This system consists as a help for the students at different aspects such as videos, meeting with professionals, simulations, flashcards and other aids.
The ***stakeholders*** of this system include:

- Students
- Teacher and Instructors
- Educational Institutions
- Content Providers
- Regulatory Bodies
- Partners and Sponsors
- Company representatives
- Developers and IT Support Team

User Characteristics:

- **Students**: Students can enroll in classes, access a variety of learning tools like flashcards, games, and simulations, attend seminars, take practice tests with performance feedback, use a Pomodoro timer for productivity, participate in Q&A hubs, manage digital textbooks, and explore educational curiosities to deepen their knowledge.
- **Content Providers**: Content providers can create and manage educational content, such as flashcards, tests, videos, and simulations; assign specific resources to students based on curriculum needs; and moderate Q&A hubs to ensure quality and relevance of content.
- **Educational Institutions**: Educational institutions are responsible for reviewing and approving or rejecting classes created under their name and can oversee requests, existing courses, and related content to maintain standards.
- **Regulatory Bodies**: Regulatory bodies review flagged content to ensure compliance with laws, approve textbook and curiosity submissions, and oversee the accuracy of educational materials and user contributions
- **Administrators**: Administrators manage and moderate past exam papers to remove low-quality or non-compliant ones and ensure that flagged content in Q&A hubs adheres to community guidelines.
- **Company Representatives**: Company representatives are tasked with adding career-related opportunities for students by creating and managing resources, such as internship listings, job boards, and career-oriented seminars or workshops.

Constraints:

1. **Scalability**: The system must handle a growing number of users and data without performance degradation.
2. **Budgetary Limitations:** The cost of development, deployment, and maintenance must align with the university's financial resources.
3. **Interoperability**: Compatibility with existing university systems (e.g., learning management systems, student portals).

Dependencies:

1. **Technological Infrastructure**: Reliable servers, databases, and network connectivity to support the system.
2. **Stakeholder Input**: Collaboration with faculty, students, and administrators to ensure the system meets their needs.
3. **Content Availability**: Dependence on educators and students to populate the platform with relevant educational materials.
4. **Maintenance and Support**: Ongoing technical support and updates to address bugs and improve functionality.

Requirements - UIS

For this software the requirements are separated based on the user level.

- For entrance to the system, if the user is a part of campus community he will enter the university email and password, and automatically the user will have access to both the UIS and the student's community.
- If the person is not part of any registered university he will have access only in the community program and it is not allowed to enter to the UIS program. If any other user tents to access it, he will get a message "Access blocked". In the community the user can enter to the software if they have payed a fee using the payment by card.

Functional Requirements

1. ***Student Registration Process***

    1. **Admission Office**
        - Collects and verifies the necessary student information for university registration.
    2. **Contract Signing & Fee Payment**
        - Student signs the contract to confirm admission.
        - **Finance Office** verifies and confirms tuition fee payment.
    3. **Registrar's Office**
        - Receives the student's information after finance approval.
        - Creates and maintains the student profile with all related data.
    4. **Email Account Creation**
        - System automatically generates a university email account for the student.
        - Email is required for academic and administrative communication throughout the academic year.

2. **Attendance Tracking Process**

    1. **Faculty Member (Lecturer/Instructor)** records attendance per session (Present, Absent, Late).
    2. **System** updates attendance percentage in real-time and flags students below the required threshold.
    3. **Student** can view their attendance breakdown by course and week.
    4. **Academic Coordinator/Department Head** monitors attendance reports and ensures compliance with university policies.
    5. **Registrar's Office** enforces attendance policies, issuing warnings or taking necessary actions based on attendance records.

3. Course Selection & Enrollment Process

1. **Registrar's Office** verifies student eligibility (contract signed, tuition paid).
2. **Finance Office** confirms fee payment and financial approval.
3. **Student** accesses the course selection system based on academic level restrictions (Freshman, Sophomore, Senior).
4. **Academic Advisor** gets a notification about the student selections, approves or rejects selected courses.
5. **Registrar's Office** finalizes enrollment, and the student's course list is updated.
6. **System** updates timetable and enables access to course materials.

4. Academic Performance & Grading Process

1. **Student** selects the academic year and semester to view grades.
2. **Faculty** enters and finalizes grades for assessments (quizzes, exams, projects).
3. **System** automatically calculates weighted final grades based on grading criteria.
4. **Professors** have the ability to apply grade curves before finalizing results.
5. **Registrar's Office** confirms and uploads final grades.
6. **ICT Office** can unlock grades for editing if corrections are needed after finalization.
7. **Student** accesses final grades in both letter and numeric format.
8. After final approval, grades are automatically recorded in the student's transcript.
9. Weighted final grades and GPA calculations are updated accordingly.

5. Document Request & Processing

1. **Student** submits a document request (transcript, certificate, etc.).
2. **Registrar's Office** reviews and processes the request, prepares the document, updating the request status.
3. **System** notifies the student when the document is ready.
4. **Student** must complete the required payment at the **Finance Office** before collecting the document.
5. **Student** collects the document in person or downloads a digital copy if available.

6. Classroom & Learning Management

1. **Student** joins a course classroom using a code provided by the professor.
2. **Faculty** uploads syllabi, lecture materials, assignments.
3. **Students & Faculty** engage through posts, announcements, and discussions.
4. **System** displays upcoming assignments and submission deadlines.

7. Professional Practice & Career Services Process

1. **Student** uploads CV and applies for available internships.
2. **Career Office** reviews applications and connects students with companies.
3. **System** tracks application status (submitted, approved, rejected).
4. **Student monitors** progress and receives updates on job placement.
5. **Career Office** can add new companies into the system, including name, logo, and description.
6. **System** maintains a company directory for student reference.

8. Student Applies for Additional Exams

1. Students submit an application for additional exams through the system.
2. **Registrar's Office** reviews the request based on university policies.
3. If approved, the additional exam is scheduled, and the student is notified.
4. Faculty conducts and evaluates the additional exam.
5. The new grade is recorded in the system.
6. **ICT Office** ensures the updated grade is reflected in the transcript.

9. Timetable Scheduling Process

1. **Admission Office** creates and finalizes the semester timetable.
2. System auto-generates schedules based on faculty availability and student enrollments.
3. **Registrar's Office** reviews the schedules to ensure they follow university policies, avoid class overlaps, and meet room capacity limits, resolving any issues found
4. **Head of department** assigns courses to the academic staff, in alignment with department requirements.
5. Classrooms and labs are allocated according to student capacity and course type.
6. **Students** access their schedules through the university portal.
7. System updates and notifies **students** and **faculty** of any schedule changes.
8. **Students** can request timetable modifications during the add/drop period
9. **Academic Advisors** and **Registrar's Office** review and approve schedule change requests.

10. Degree Audit Process

1. **Registrar's Office** retrieves student academic records.
2. **System** analyzes completed courses, earned credits, and remaining requirements.
3. **System** generates a degree audit report, displaying progress toward graduation.
4. **Registrar's Office** verifies if the student meets graduation eligibility criteria.
5. **System** highlights missing courses or credits if requirements are unmet.
6. **Student** receives notifications regarding degree progress and remaining coursework.
7. **Registrar's Office** updates the audit status once graduation requirements are fulfilled

Nonfunctional Requirements

- **Product Requirements**

These define how the product must behave, focusing on usability, performance, reliability, and functionality.

- **Usability Requirements**

1. An intuitive interface for students, faculty, and staff to ensure smooth navigation across functions (e.g., course selection, grading, financial information access) by using descriptive names and easy interfaces.

- **Performance Requirements**

2. The system must handle up to 5,000 concurrent users during peak times without delays (more than 2 seconds) or crashes.
3. Response time for actions like grade updates or course selections must be less than 2 seconds.
4. Real-time updates shall be provided for student and financial information, ensuring no duplication of data.
5. The system shall accommodate increasing numbers of users and additional university departments without a reduction in performance.

- **Organizational Requirements**

These arise from organizational processes, policies, or internal standards.

6. Users shall authenticate their access to the system through two forms of verification: primary authentication(A valid username and password); secondary authentication(A one-time passcode (OTP) sent via email, SMS, or a designated authenticator app.)
7. Restrict access to the University Information System (UIS) to users with verified school accounts (no external or unauthorized accounts allowed).
8. Sensitive information, such as financial and personal data, must be encrypted both at rest and in transit.
9. The registration process for students shall not take longer than 10 minutes to complete.
10. The system shall support up to 2,000 student registrations per day.
11. Faculty shall submit grades and attendance updates for each student within a maximum of 2 weeks after classes conclude; updates beyond this period shall not be permitted.
12. The system shall guarantee 99.9% availability, with automated backups and disaster recovery protocols in place.

- **External Requirements**

These result from external factors such as legal, regulatory, or cultural considerations.

13. The student registration must be held within the registration period specified by the Educational Ministry.

Requirements - ThinkLink

Functional Requirements

1. Class Enrollment & Access Process (Student & System)

1. The **system** displays available and enrolled classes for students.
2. A **student** selects a class and clicks "Enroll."
3. The **system** updates the student's enrollment status and grants access to class materials.
4. Only enrolled **students** gain access to interactive features (tests, flashcards, games, and seminars).
5. The **system** enforces restrictions for non-enrolled students (viewing only materials but no participation).

2. Class Creation & University Approval (Content Provider & University Representative)

1. A **content provider** creates a new class and assigns it to a university.
2. **The university** receives an automatic notification of the new class request.
3. **The university representative** approves or rejects the request.
4. If approved, the class is added to the university's list of official courses.
5. If rejected, the content provider receives a rejection notification with feedback.

3. Test & Seminar Participation (Student & Content Provider)

1. A **content provider** creates a test or schedules a seminar.
2. Enrolled **students** receive a notification about the test or seminar.
3. **Students** access and participate in the test or seminar at the scheduled time.
4. The **system** records test responses and seminar attendance.
5. The **content provider** reviews test results and seminar participation data.
6. **Students** receive feedback on their test performance

4. Curiosities:

1. **Users** can create and submit their own curiosities by sharing educational facts, insights, or thought-provoking questions. They can add relevant tags, select appropriate categories, and provide a detailed description for each submission.
2. **The system** allows users to browse, explore, and discover educational facts, curiosities, and insights across various academic fields. Content is categorized by subject, trending topics, and user preferences, ensuring a personalized learning experience.
3. **Users** can engage with curiosities by liking, commenting, and sharing them within the platform. Discussion forums enable deeper academic debates, while a bookmarking feature allows users to save curiosities for later reference.

5. Simulations and Video Explanations Process

1. **Student** explores complex concepts through interactive simulations in a virtual environment.
2. **System** allows students to pause, rewind, and take notes within the video interface for better comprehension.
3. **Platform** provides video explanations for key topics, created by subject-matter experts.
4. **System** recommends related simulations and video explanations based on students' learning progress by analyzing their past interactions, quiz results, frequently accessed topics, and areas where they need improvement.
5. **Content Providers** assign specific simulations and video lessons for a specific category for all students.

6. Practice tests and Quizzes:

1. **Student takes** timed and untimed practice tests with various question types, including multiple-choice, short answer, and essay-style questions.
2. **System** provides auto-grading and instant feedback to enhance learning.
3. **System** generates performance reports and analytics to help students identify strengths and weaknesses.
4. **Student** monitors progress, reviews detailed explanations of correct answers, and revisits previous exam attempts.
5. **Content Provider** clicks the "Create Test" button to design multiple questions and uploads the test to the system with its name specified.

Q&A Hub Processes

7. Create, Answer, and Interact with posts

1. **User** logs in to the system.
2. **User** creates a new post or question by entering a title, category, description, and tags.
3. **User** submits the post, and the system displays it in the relevant category.
4. Other **users** can view the post and respond by submitting answers or comments.
5. **Users** can upvote or downvote both questions and answers based on usefulness and accuracy.
6. The **original poster** can mark a response as the "Best Answer" to highlight the most helpful solution.
7. **System** sends notifications when users receive responses, comments, or mentions.

8. Content Moderation and Management

1. **User** flags inappropriate, spam, or off-topic content.
2. **Moderators** receive notifications of flagged content.
3. **Moderators** can edit, hide, or remove content that violates guidelines.
4. **Content Providers and Academic Staff** can also edit or delete inappropriate posts.

9. Search and Discover Content

1. **User** enters keywords or phrases into the search bar.
2. **System** retrieves relevant questions, answers, and discussions.
3. **User** browses the results and selects the most relevant content.

10. Set Up and CustomizePomodoro Timer

1. **User** navigates to the Pomodoro Timer feature.
2. **User** configures settings, including work session length, break duration, and cycle count.
3. **User** starts the timer, and the system begins tracking the session.
4. The **system** notifies the user when the work session ends and break time starts.

11.Task Management and Progress Tracking

1. **User** creates a list of tasks to complete during sessions.
2. **User** edits or deletes tasks as needed.
3. The **system** tracks completed tasks and displays progress.
4. Notifications remind users to start new sessions if inactive.

12. Flashcard Creation & Management

1. A **content provider** creates flashcards by adding text, images, or mathematical formulas.
2. The **system** saves the flashcards and allows organization into categorized decks.
3. The **content provider** can edit, update, or delete flashcards as needed.

13.Flashcard Access & Study Modes

1. A **student** browses or searches for flashcard decks using keywords, tags, or subjects.
2. The **system** displays available flashcards, and the student selects a study mode:
   - Standard Mode (flip front and back of cards).
   - Quiz Mode (multiple-choice or fill-in-the-blank).
   - Spaced Repetition Mode (adaptive review based on performance).
3. The **student** can shuffle cards for randomized practice.

14.Search, Tracking & Gamification

1. A **student** searches for flashcards using keywords, tags, or subjects.
2. The **system** uses the Spaced Repetition [1] to suggest flashcards that need revision based on past performance.
3. The **system** tracks progress, providing mastery analytics and revision suggestions.
4. **Students** earn points and badges for completing revision tasks.
5. **Students** can participate in weekly/monthly challenges where they compete on quiz scores or streaks to earn additional rewards.

[1] Spaced repetition is a special algorithm which helps decide when to show a flashcard again based on how well the student remembers it. If they struggle, it appears sooner; if they remember well, it appears later.

15.Job and Internship Listings

1. A **company representative** posts job and internship listings on the platform, providing details like company name, role description, and application deadlines.
2. The **company representative** categorizes listings by industry, location, and job type (full-time, part-time, remote).
3. **Students** search using filters (industry, location, job type), and the system displays relevant results.

16. Profile Creation & Customization

1. A **student** creates a career profile, entering details about their education, skills, and work experience.
2. The **student** can edit their profile at any time to update relevant information.

17. Search & Application Process

1. A **student** searches for internships or job opportunities using keywords, location, job type, or experience level.
2. The **student** selects a job or internship and applies directly by attaching relevant documents (e.g., resume, cover letter).
3. The **company representatives** can shortlist candidates, contact them for interviews, or reject applications.
4. The system updates the application status (e.g., shortlisted, interview scheduled, rejected) and notifies students accordingly.

18. Virtual Career Fairs & Events

1. **Company representatives** host virtual career fairs or webinars to interact with students.
2. **Students** attend these events to learn more about job opportunities and network with company representatives.

19. Past Papers Process

1. **Student** uploads, categorizes, and searches past exam papers, quizzes, and assignments based on university, course, subject, and academic year.
2. **System** allows students to preview past papers before downloading and enables them to rate, review, and provide feedback to ensure quality and relevance.
3. **Administrators** use moderation tools to review and remove past papers that violate content guidelines or are of low quality.

20. Textbooks

1. **System** provides a digital library where students can browse and search for textbooks categorized by subject, course, and academic year.
1. **Users** can download textbooks in multiple formats, such as PDF, and access them through an integrated reader with highlighting, annotations, and bookmarks.
2. **System** enables users to translate textbooks into different languages and customize reading modes for better usability.
3. **Content Providers** upload and maintain educational materials through textbook management.
1. **Regulatory Bodies** oversee approvals to ensure compliance with academic standards.

21. Seminars Process

1. **Students and Institutions** create and manage seminars, including details such as topic, speaker(s), date, time, and delivery mode.
2. **Students** register for seminars and receive automated reminders via email or in-app notifications.
3. Online Seminars support interactive features such as live chat, Q&A sessions, and polls to enhance engagement.
4. **System** stores recorded seminars, allowing students to access and replay them later for reference.

Nonfunctional Requirements

- **Product Requirements**

These specify how the product must behave, including usability, performance, availability, and security requirements.

- **Usability Requirements**

1. The platform must offer a user-friendly and visually appealing interface for students, teachers, and content providers to ensure smooth navigation across functions such as course management, collaboration, and content delivery.
   - All core features, such as course creation, grading, and video conferencing, shall be accessible from the homepage within 3 clicks or fewer.
   - Interface components (e.g., buttons, menus) must respond to user actions (e.g., clicks, taps) within 1 second.

- **Performance Requirements**

3. The system must support up to 10,000 active users at any given time without delays, crashes, or degraded performance.
4. Loading time for videos, flashcards, and simulations must not exceed 3 seconds.
5. The system must seamlessly scale to accommodate:
   - A 10% annual growth in the number of users.
   - Addition of at least 50 new courses per quarter and 3 new institutions per year without impacting system performance.

6. The platform shall maintain optimal performance with an average response time for core actions (e.g., enrolling in courses, submitting assignments) not exceeding 2 seconds, even under increased usage.

- **Availability Requirements**

7. The community system must ensure 24/7 availability for students worldwide, with redundancy solutions to prevent downtime.
8. Automated failover systems and disaster recovery mechanisms shall ensure data is restored within 15 minutes in case of a system outage.
9. A 99.9% uptime guarantee shall be enforced, meaning downtime shall not exceed 8.76 hours annually. Scheduled maintenance shall occur during off-peak hours (e.g., 2 AM–5 AM UTC) and must not exceed 4 hours per month.

- **Security Requirements**

10. For the Student Community, users must register an account specifically for the platform. Access will require payment of a fee, after which users can log in with their registered account.

- **Organizational Requirements**

These arise from organizational processes, policies, or internal standards.

11. The platform must support a minimum of 10 languages (e.g., English, Spanish, Chinese) and allow users to select their preferred regional settings (e.g., time zone, currency format) during account setup.
12. Language translations must not exceed a 2-second delay during content rendering (e.g., interface text and course descriptions).

- **External Requirements**

These result from external factors such as legal, regulatory, or cultural considerations.

13. The platform shall actively monitor and prevent the dissemination of harmful, discriminatory, or offensive content, ensuring a safe and inclusive environment that encourages collaboration and mutual support among students.

UIS & Students Community

Nonfunctional Requirements for both

- **Product Requirements**
  These focus on how the product must behave, including functionality, usability, and performance:

  1. The software must allow easy updates, bug fixes, and feature enhancements to meet the evolving needs of users through continuous refactoring.
  2. Both systems must be accessible via web browsers and mobile devices (iOS/Android) with full functionality on each platform.

- **External Requirements**
  These are driven by external factors, such as legal, regulatory, or regional constraints:

  3. The platform shall comply with the Albanian Law "For the Protection of Personal Data," ensuring that students' personal and academic information is securely stored, processed, and accessed only for educational purposes.

Use Case

UC Name	***UIS -UC-1***
Summary	*(Students Registration Process): This requirement depicts the registration of a student in the system by the university.*
Dependency	*No dependencies*
Actors	***Student**, Admission Office, Finance Office, Registrar's Office*
Preconditions	*The matura students data and the required documents (matura diploma) must be available to the Admission Office.*
Description of the Main Sequence	• ***Step 1**:The admission office verifies the necessary student information for university registration.* • ***Step 2:** The admission office prepare the contract.* • ***Step 3:** Student signs the contract to confirm admission.* • ***Step 4:** Finance Office verifies tuition fee payment.* • ***Step 5:** Finance Office confirms tuition fee payment.* • ***Step 6:** Registrar Office receives the student's information after finance approval.* • ***Step 7:** System automatically generates a university email account for the student.* • ***Step 8:** Registrar Office creates the students profiles.* • ***Step 9:** Registrar Office maintains the students profiles.*
Description of the Alternative Sequence	• ***Step 5:** Finance office may reject or approve the tuition fee payment. If rejected the registration process interrupts.*
Non functional requirements	*The registration process will not require more than 10 minutes.* *The system must be able to handle up to 2000 students registrations per day, within the registration period specified by the Educational Ministry.*
Postconditions	*The student is registered successfully to the university and the email and the profile is created.*

Use Case

UC Name	***UIS -UC-2***
Summary	*(Course Selection & Enrollment Process): This requirement depicts the registration of a student in the system by the university.*
Dependency	*UIS-US-1*
Actors	***Student(Freshman, Sophomores, Seniors)**, Finance Office, Registrar's Office, Advisor*
Preconditions	*The student must sign the contract, pay the tuition fee, and must be the registered into the system.*
Description of the Main Sequence	• **Step 1:** Registrar's Office verifies student eligibility (contract signed, tuition paid). • **Step 2:** Finance Office confirms fee payment and financial approval. • **Step 3:** Student accesses the course selection system based on academic level restrictions (Freshman, Sophomore, Senior) by clicking on "Course Selection" field. • **Step 4:** Student select the courses and click submit. • **Step 5:** Student confirm the selection. • **Step 6:** Academic Advisor gets a notification about the student selections. • **Step 7:** Academic Advisor approves or rejects selected courses. • **Step 8:** Registrar's Office finalizes enrollment. • **Step 9:** System updates the student's course list. • **Step 10:** System updates timetable and enables access to course materials.

Description of the Alternative Sequence	• ***Step 4:***
	  ○ *The freshman must be able to select only the courses available for the respective semester.*
	  ○ *The sophomores and seniors must be able to select courses that belongs to the respective semester and the last year semester. They must first select the failed courses and then the other.*

Non functional requirements	*The course must be selected only within the period specified on the university policies.*
Postconditions	*The student course selection is completed.*

Use Case

UC Name	***UIS-UC-3***
Summary	*This use case describes the process of tracking student attendance, including recording attendance, updating records, and monitoring attendance compliance.*
Dependency	*UIS-US-1, UIS-US-2*
Actors	***Student,*** *Faculty Member, Academic Coordinator/Department Head, Registrar's Office*
Preconditions	*The student must be enrolled in a course, and the university system should be active for attendance tracking.*
Description of the Main Sequence	• **Step 1:** *The faculty member records each student's status as Present, Absent, or Late for the session.* • **Step 2:** *The system calculates and updates the attendance percentage for each student in real time based on the session data.* • **Step 3:** *The student accesses the system to view their attendance breakdown by course and week.* • **Step 4:** *The system generates the attendance report.* • **Step 5:** *The Academic Coordinator or Department Head accesses attendance reports to monitor compliance with university attendance policies.* • **Step 6:** *The Registrar's Office reviews attendance records.* • **Step 7:** *The Registrar's Office takes necessary actions, such as issuing warnings or other measures, for students with low attendance.*
Description of the Alternative Sequence	**Step 3:** *If a student is incorrectly marked as absent, they can submit a correction request, which must be verified by the Faculty Member.* **Step 7:** o *If a student's attendance falls below 75%, they will receive a notification via email.* o *Students with attendance below 75% will not be allowed to attend the midterm or final exams.*
Non-functional Requirements	*The attendance system should update in real time and provide accurate reports. It must handle large-scale student data efficiently without delays.*
Postconditions	*Attendance records are updated, and students below the threshold are flagged for necessary actions. The system ensures transparency in attendance monitoring.*

Use Case

UC Name	***UIS -UC-4***
Summary	*(Academic Performance & Grading Process): This requirement depicts grading part of the students.*
Dependency	*UIS-US-1, UIS-US-2*
Actors	***Student,*** *Faculty, Registrar's Office, ICT Office*
Preconditions	*The student must be register to the university and have already selected the courses*
Description of the Main Sequence	• **Step 1:** Faculty enters grades for assessments (quizzes, exams, projects). • **Step 2:** Faculty finalizes grades for the courses. • **Step 3:** System automatically calculates weighted final grades based on grading criteria. • **Step 4:** Registrar's Office confirms and uploads final grades. • **Step 5:** Student accesses final grades in both letter and numeric format. • **Step 6:** After final approval, grades are automatically recorded in the student's transcript. • **Step 7:** Weighted final grades and GPA calculations are updated accordingly.

Description of the Alternative Sequence	• ***Step 2:***
	  ○ *Professors have the ability to apply grade curves before finalizing results.*
	  ○ *ICT Office can unlock grades for editing if corrections are needed after finalization.*
	• ***Step 5:*** *By default, students can view grades solely for their current semester courses, but they have the option to apply filters for the academic year and semester if needed.*

Non functional requirements	*The grading system should update in real time and provide accurate reports. Weighted final grades and GPA calculations re done based on the policies of the university.*
Postconditions	*The student can have access to see their grades.*

Use Case

UC Name	***UIS -UC-5***
Summary	*(Document Request & Processing): This requirement describes the process of requesting, processing, and delivering official documents such as transcripts and certificates.*
Dependency	*UIS-UC-1*
Actors	***Student**, Finance Office, Registrar's Office*
Preconditions	*The student must have an active university account. The requested document must be available in the system.*
Description of the Main Sequence	• ***Step 1:** Student click on "Document Request" field.* • ***Step 2:** Student submits a document request (e.g., transcript, certificate) through the system.* • ***Step 3:** Registrar's Office reviews the request.* • ***Step 4:** Registrar's Office prepare the document* • ***Step 5:** Registrar's Office update the request status.* • ***Step 6:** The system notifies the student when the document is ready for collection.* • ***Step 7:** Student completes the required payment at the Finance Office.* • ***Step 8:** The system provides the document with an electronic seal for authenticity.* • ***Step 9:** Student collects the document in person or downloads a digital copy if available.* • ***Step 10:** The system updates the document request status to completed**.***
Description of the Alternative Sequence	• ***Step 7:** If the Finance Office rejects the payment or any mistake is done, the request remains pending until the payment is successfully completed.*
Non functional requirements	*The processing time for document requests should not exceed* **24 hours** *(excluding weekends and holidays).* *The system must handle at least **500 document requests per day** without performance degradation.*
Postconditions	*The student successfully receives the requested document, either in physical or digital format.* *Document request status is updated completed**.***

Use Case

UC Name	***UIS -UC-6***
Summary	*(Classroom & Learning Management): This requirement depicts the process where students enroll in courses using a code provided by the professor, to access course materials.*
Dependency	*UIS-UC-1, UIS-UC-2*
Actors	***Student**, Faculty (Academic Staff)*
Preconditions	*The faculty must have created and set up the course in the system, and the course code is automatically generated.*
Description of the Main Sequence	• ***Step 1:** The student enters the code (provided by professor) into the system to join the course.* • ***Step 2:** The faculty uploads course materials (syllabi, lectures, assignments), announcements and discussion topics in the system.* • ***Step 3:** The student accesses the uploaded materials, participates in discussions and interacts with course content.* • ***Step 4:** The faculty posts assignments with the specified deadline.* • ***Step 5:** The student submits assignments through the system before the deadline.* • ***Step 6:** The faculty reviews the submissions* • ***Step 7:** The faculty provides feedback in the system.*
Description of the Alternative Sequence	• ***Step 2:** If the student enters an invalid or expired enrollment code, the system will notify the student that the code is incorrect, otherwise he will enroll to the system.* • ***Step 5:** If the deadline is within the current week show it on the left of the announcements, otherwise do not.*
Non functional requirements	*Upload and download times for materials (e.g., syllabi, assignments) should not exceed 3 seconds under normal conditions.* *Real-time updates shall be provided for student information.*
Postconditions	*The student is enrolled in the course and has access to all course materials, including syllabi, lecture notes, and assignments*

Use Case

UC Name	***UIS -UC-7***
Summary	***Professional Practice & Career Services Process:*** *This requirement outlines the management of student internship applications, status tracking, and company connections by the Career Office.*
Dependency	*UIS-UC-1*
Actors	***Student ,*** *Career Office, Company, System*
Preconditions	*Student (Sophomore, Seniors) must have selected the professional practice course.*
Description of the Main Sequence	• ***Step 1****: Student uploads their CV.* • ***Step 2:*** *Student search for internship based on the position and company.* • ***Step 3:*** *Student double click on the internship position and see the details.* • ***Step 4:*** *Student apply for a certain position by clicking on the apply button.* • ***Step 5:*** *Career Office reviews applications.* • ***Step 6:*** *Career Office evaluates student eligibility.* • ***Step 7:*** *Career Office connects students with companies.* • ***Step 8:*** *Application status are saved (submitted, approved, or rejected)* • ***Step 9:*** *System maintains a company directory off internship companies and available positions.*
Description of the Alternative Sequence	***Step 6:*** *If the Career Office confirms the students eligibility, it connects the student with the company, otherwise the application is rejected.* ***Step 4:*** ○ *If the student is approved for internship, the Career Office updates the number of available positions in the selected company.* ○ *If any existing company offer any new vacant position, it must me reflected.* ○ *If any new company is added it must be shown in the system.*
Non-functional requirements	*The system should be able to handle up to 1,000 concurrent students applying for internships.*
Postconditions	*The student application status is updated successfully.*

Use Case

UC Name	***UIS -UC-8***
Summary	*(Student Applies for Additional Exams): This requirement outlines the process for students to apply for additional exams.*
Dependency	*UIS-UC-1, UIS-UC-3, UIS-UC-4*
Actors	***Student**, Registrar's Office, Faculty, ICT Office*
Preconditions	*The student must be enrolled in the course and eligible to apply for an additional exam according to university policies.*
Description of the Main Sequence	• ***Step 1:** Students submit an application for an additional exam.* • ***Step 2:** The system forwards the application to the Registrar's Office for review.* • ***Step 3:** The Registrar's Office verifies eligibility and request validity.* • ***Step 4:** If approved, the Registrar schedules the exam.* • ***Step 5:** The system notifies the student and respective faculty.* • ***Step 6:** After the exam was conducted, the Faculty records the results.* • ***Step 7:** The system updates the grades.* • ***Step 8:** The ICT Office ensures the grade is reflected in the student's transcript.*
Description of the Alternative Sequence	***Step 3:** If the student's application does not meet the university's policies or eligibility criteria, the Registrar's Office rejects the request. The system notifies the student of the rejection.*
Non functional requirements	*The system must process and forward an additional exam application within 5 seconds after submission.* *The Registrar's Office should be able to approve or reject applications within 2 business days.* *The system should support up to 500 concurrent exam applications without performance degradation.*
Postconditions	*The student's grade for the additional exam has been successfully recorded in the system and reflected in their transcript.*

Use Case

UC Name	***UIS -UC-9***
Summary	*(Timetable Management Process): This requirement depicts the process of creating and managing the university timetable.*
Dependency	*UIS-UC-1, UIS-UC-2*
Actors	***Students**, Head of Department, Registrar's Office, Faculty Members,*
Preconditions	*Faculty availability, course requirements, and classroom data must be available to the Head of Department.*
Description of the Main Sequence	• ***Step 1:** Head of Department sees and verifies data related to courses, faculty availability, and classroom capacity.* • ***Step 2:** He creates the initial timetable based on collected data.* • ***Step 3:** Faculty members review and propose adjustments if necessary.* • ***Step 4:** Head of Department finalizes the timetable and submits it to the Registrar's Office for approval.* • ***Step 5:** Registrar's Office reviews the timetable.* • ***Step 6:** Registrar's Office approves the timetable.* • ***Step 7:** System publishes the finalized timetable to students and faculty members through the portal.* • ***Step 8:** System sends automated notifications to all relevant users.*
Description of the Alternative Sequence	• ***Step 6:** Registrar's Office may request changes or reject the timetable. In case of rejection, the Head of Department revises and resubmits the timetable.*
Non functional requirements	*The timetable creation and publishing process should not take more than 30 minutes. The system must support up to 100 timetable changes per semester.*
Postconditions	*The finalized timetable is successfully published and accessible to students and faculty members.*

Use Case

UC Name	***UIS -UC-10***
Summary	*(Degree Audit Process): This requirement depicts the process of verifying a student's academic progress towards graduation.*
Dependency	*UIS-UC-1*
Actors	*Seniors Student, Registrar's Office, System*
Preconditions	*The student's academic records must be available and up-to-date in the system.*
Description of the Main Sequence	• ***Step 1:*** *Registrar's Office retrieves student academic records.* • ***Step 2:*** *System analyzes completed courses, earned credits, and remaining requirements.* • ***Step 3:*** *System generates a degree audit report, displaying progress toward graduation.* • ***Step 4:*** *Registrar's Office verifies if the student meets graduation eligibility criteria.* • ***Step 5:*** *System highlights missing courses or credits if requirements are unmet.* • ***Step 6:*** *Student receives notifications regarding degree progress and remaining coursework.* • ***Step 7:*** *Registrar's Office updates the audit status once graduation requirements are fulfilled.*
Description of the Alternative Sequence	• ***Step 4:*** *If graduation eligibility criteria are not met, the system sends a notification of missing requirements to the student and the audit status remains pending.*
Non functional requirements	*-The degree audit process should not exceed 3 minutes per student. -The system must support up to 1500 degree audit processes per day during peak periods.*
Postconditions	*A degree audit report is generated, and the student is notified of their progress. The audit status is updated based on the fulfillment of graduation requirements.*

Use Case

UC Name	***ThinkLink-UC-1***
Summary	*(Class Creation & University Approval): This requirement describes how a content provider creates a new class and seeks university approval.*
Dependency	*No dependencies*
Actors	***Content Provider**, University Representative*
Preconditions	*The content provider must have the necessary permissions to create a class.*
Description of the Main Sequence	• ***Step 1**: The content provider creates a new class and specifies the university.* • ***Step 2:** The system generates a notification to the university representative about the new class request.* • ***Step 3**: The university representative reviews the request* • ***Step 4**: The university representatives either approves or rejects the request.* • ***Step 5**: If approved, the class is added to the university's list of official courses.* • ***Step 6:** The system notifies the content provider of the decision.*
Description of the Alternative Sequence	• ***Step 3:** If the university rejects the request, the system notifies the content provider with feedback for potential resubmission.*
Non functional requirements	*The university should process class approval within 48 hours.* *The system should notify users of approval/rejection within 2 sec after processing.*
Postconditions	*If approved, the class is available for students to enroll in. If rejected, the content provider can review the feedback and resubmit the request.*

Use Case

UC Name	***ThinkLink-UC-2***
Summary	*(Class Enrollment & Access Process): This requirement describes the process of students enrolling in classes and gaining access to course materials.*
Dependency	*ThinkLink-UC-1*
Actors	***Student***
Preconditions	*The student must have an active university profile and be logged into the system. The classes must be created.*
Description of the Main Sequence	• ***Step 1****: The system displays available and enrolled classes for students.* • ***Step 2:*** *A student chooses an unenrolled class.* • ***Step 3:*** *The system grants access to class materials of selected class.* • ***Step 4****: The student click on "Enroll" button.* • ***Step 5****: The system updates the student's enrollment status.* • ***Step 6:*** *The student can participate in interactive features such as tests, flashcards, games, and seminars.*
Description of the Alternative Sequence	• ***Step 5:*** *If the class has reached maximum capacity, the system displays a notification, and enrollment is not processed.* • ***Step 6:*** *If the student is not enrolled in the class it can only see the course materials but not participate on interactive features.*
Non functional requirements	*The system should process an enrollment request in under 2 seconds. The system must support up to 500 concurrent enrollments at a given time.*
Postconditions	*The student is successfully enrolled in the selected class and has access to all permitted materials and activities.*

Use Case

UC Name	***ThinkLink-UC-3***
Summary	*(Test & Seminar Participation): This requirement covers student participation in tests and seminars.*
Dependency	*ThinkLink-UC-1, ThinkLink-UC-2*
Actors	***Student,*** *Content Provider*
Preconditions	*A test or seminar must be scheduled, and students must be enrolled in the class.*
Description of the Main Sequence	• ***Step 1****: The content provider creates a test (with different types of questions) or schedules a seminar.* • ***Step 2:*** *The system automatically notifies enrolled students about the test or seminar.* • ***Step 3****: Students log in and participate at the scheduled time.* • ***Step 4****: The system records test responses and seminar attendance, after each completion.* • ***Step 5:*** *The content provider reviews participation data and test results.* • ***Step 6:*** *Students receive feedback on their performance.*
Description of the Alternative Sequence	***Step 1:*** *If for any reason the test or seminar will not be held, It can be cancelled with a specified reason, and the student will get an notification.* ***Step 3:*** o *If a student misses the test/seminar, the system marks them as absent.* o *If the student participates on a test, he may select to attend the test in different format: multiple choice questions, open questions, true or false, or he may want to combine them.*
Non functional requirements	*The system should support at least 1000 simultaneous test participants.* *Test results should be available to students within 24 hours of submission.*
Postconditions	*The test results and seminar participation records are stored, and students receive feedback.*

Use Case

UC Name	***ThinkLink -UC-4***
Summary	*(Curiosities): This requirement depicts a system that allows users to share, discover, engage with, and interact with educational content, promoting learning, discussions, and personal exploration across different subjects.*
Dependency	*ThinkLink-UC-1, ThinkLink-UC-8, ThinkLink-UC-9*
Actors	***User***
Preconditions	*The user is logged into the platform and has access to the main dashboard where they can browse, submit, and interact with curiosities.*
Description of the Main Sequence	• ***Step1:*** *After the user logs into the platform, he selects a category of interest.* • ***Step 2:*** *The system displays related curiosities, facts, and insights.* • ***Step 3:*** *The user clicks on a curiosity to view its full details* • ***Step 4:*** *The user engages with the curiosity (like, comment, share).* • ***Step 5:*** *The system records engagement and updates interaction metrics.* • ***Step 6:*** *The user bookmarks the curiosity for future reference.*
Description of the Alternative Sequence	• ***Step 3:*** o *If no curiosities are found in the selected category or topic, the system notifies the user with a message stating that no content is available in that category and offers suggestions to explore other categories or trending topics.* o *The user can submit their own curiosity, which is reviewed and added to the platform if validated.*
Non functional requirements	*The system must support up to 10,000 active users at any given time without delays, crashes, or degraded performance.*
Postcondition	*The user has successfully interacted with the system, having either engaged with curiosities, submitted a new curiosity, or explored different categories and topics, with all actions and preferences recorded in their account for future reference.*

Use Case

UC Name	**ThinkLink-UC-5**
Summary	(Simulations and Video Explanations Process**):** This requirement outlines the use of interactive simulations and video explanations to enhance student learning through a virtual environment.
Dependency	*ThinkLink-UC-1, ThinkLink-UC-2*
Actors	***Student,*** *System****,*** *Content providers*
Preconditions	Students must have access to the platform and a registered account to interact with simulations and videos.
Description of the Main Sequence	• **Step 1:** The Content Provider assigns specific simulations relevant to the class curriculum and curated video lessons aligned with learning objectives to be accessed by students. • **Step 2**: Student search the topic of the interest. • **Step 3**: System provides video explanations for key topics, created by subject-matter experts. • **Step 4**: Student watches these videos for deeper understanding. • **Step 5**: Student engage with simulations through pausing, rewinding, and taking notes within the video interface for better comprehension.
Description of the Alternative Sequence	• ***Step 2:*** o Students explore new topics or select the simulations and video explanations, recommended by the system, based on students' learning progress by analyzing their past interactions, quiz results, frequently accessed topics, and areas where they need improvement. o Student reviews past simulations and video explanations to reinforce learning without engaging in new recommendations.
Non-functional requirements	The system should be able to support real-time interactions within simulations for up to 5,000 concurrent users.
Postconditions	*The student progress is updated*

Use Case

UC Name	***ThinkLink-UC-6***
Summary	*(Practice Tests and Quizzes): This requirement describes the process of students taking practice tests and quizzes, receiving feedback, and analyzing their performance, as well as the role of Content Providers in creating and managing tests.*
Dependency	*No dependencies*
Actors	***Student**, Content Provider*
Preconditions	The students must have been signed in the system to have access.
Description of the Main Sequence	• ***Step 1:** Content Provider clicks the **"Create Test"** button.* • ***Step 2:** Content Provider designs multiple questions.* • ***Step 3:** Content Provider uploads the test to the system with a specified name.* • ***Step 4:** Student selects a practice test and chooses between timed or untimed mode.* • ***Step 5:** The system presents various question types, including multiple-choice, short answer, and essay-style questions.* • ***Step 6:** The system provides auto-grading for objective questions.* • ***Step 7:** The system generates performance reports and analytics, highlighting the student's strengths and weaknesses.* • ***Step 8:** The system delivers instant feedback on performance.* • ***Step 9:** Student reviews detailed explanations for correct answers and can revisit past exam attempts.*
Description of the Alternative Sequence	*N/A (The process is continuous and allows updates anytime.)*
Non functional requirements	*The system must provide **instant feedback** for all auto-gradable questions within **5 seconds** after submission.* *The system should support **at least 10,000 simultaneous test attempts** without performance issues.* *Performance reports should be **generated within 5 seconds** after test completion.*
Postconditions	*The student successfully completes the test and receives **performance insights.*** *The system stores test attempts for **future review and tracking.***

Use Case

UC Name	***ThinkLink-UC-7***
Summary	*(Create, Answer, and Interact with Posts): This requirement depicts the process of creating, answering, and interacting with posts on the ThinkLink platform.*
Dependency	*No dependencies*
Actors	***User*** *(Student or Academic Staff), System*
Preconditions	*The user must be logged into the system with an active profile.*
Description of the Main Sequence	• ***Step 1:*** *User creates a new post or question by entering a title, category, description, and tags.* • ***Step 2:*** *User submits the post.* • ***Step 3:*** *The system displays the post in the relevant category.* • ***Step 4:*** *Other users view the post and respond by submitting answers or comments.* • ***Step 5:*** *Users upvote or downvote questions and answers based on usefulness and accuracy.* • ***Step 6:*** *The original poster marks a response as the "Best Answer" to highlight the most helpful solution.* • ***Step 7****: System sends notifications when users receive responses, comments, or mentions.*
Description of the Alternative Sequence	• ***Step 3:*** *If the post format is incorrect or missing required fields, the system prompts the user to complete the necessary information*
Non functional requirements	*Post creation and interaction processes should not exceed 3 seconds. The system must handle up to 100,000 interactions per day.*
Postconditions	*The post is successfully created, and users are notified of interactions and responses.*

Use Case

UC Name	***ThinkLink -UC-8***
Summary	*(Content Moderation and Management): This requirement depicts the process of flagging and managing inappropriate or spam content on the ThinkLink platform.*
Dependency	*No dependencies*
Actors	***Student,*** *Content Provider, Academic Staff, System* ***Moderators can be either Content Providers or Academic Staff*
Preconditions	*The system must have moderation tools and guideline policies in place.*
Description of the Main Sequence	• ***Step 1:*** *User flags inappropriate, spam, or off-topic content.* • ***Step 2:*** *System notifies moderators of the flagged content.* • ***Step 3:*** *Moderators review the flagged content.* • ***Step 4:*** *Moderators verify whether it violates guidelines.* • ***Step 5:*** *Moderators can edit, hide, or remove content that violates guidelines.* • ***Step 6:*** *Content Providers and Academic Staff can also edit or delete inappropriate posts.* • ***Step 7:*** *System logs all moderation actions for auditing.*
Description of the Alternative Sequence	• ***Step 4:*** *If the content does not violate guidelines, the moderator dismisses the flag and restores the content's visibility.*
Non functional requirements	*Moderation actions should be performed within 5 minutes of flagging. The system should support up to 5,000 moderation actions per day.*
Postconditions	*The content is either approved, edited, hidden, or removed, and the action is logged.*

Use Case

UC Name	**ThinkLink-UC-9**
Summary	(Search and Discover Content Process**)** This requirement outlines how users can search for and discover relevant questions, answers, and discussions using the platform.
Dependency	*No dependencies*
Actors	***Users,*** *System*
Preconditions	Users must have access to the platform
Description of the Main Sequence	• **Step 1**: User enters keywords or phrases into the search bar • **Step 2**: System retrieves relevant questions, answers, and discussions, based on the keyword. • **Step 3**: User browses the results • **Step 4**: Users selects the most relevant content
Description of the Alternative Sequence	• ***Step 1:*** If the search results are not satisfactory, the user can enter a different keyword or phrase.
Non-functional requirements	The system should deliver search results within **2 seconds** for optimal user experience.
Postconditions	User successfully finds relevant content, refines the search, or explores alternative recommendations.

Use Case

UC Name	***ThinkLink -UC-10***
Summary	*(Set Up and Customize Pomodoro Timer): This requirement depicts the process of configuring and using the Pomodoro Timer for efficient time management.*
Dependency	*No dependencies*
Actors	***User,*** *System*
Preconditions	*The user must be logged into the system and have access to the Pomodoro Timer feature.*
Description of the Main Sequence	• ***Step 1:*** *User navigates to the Pomodoro Timer feature.* • ***Step 2:*** *User configures settings, including work session length, break duration, and cycle count.* • ***Step 3:*** *User starts the timer.* • ***Step 4:*** *The system begins tracking the session.* • ***Step 5:*** *System notifies the user when the work session ends and break time starts.* • ***Step 6:*** *System tracks the number of completed cycles and total time spent.*
Description of the Alternative Sequence	• ***Step 2:*** *If the user does not customize settings, the system uses default values (e.g., 25 minutes work, 5 minutes break).*
Non functional requirements	*The timer should have high accuracy with less than a 1-second delay. Notifications should be prompt and non-intrusive.*
Postconditions	*The Pomodoro Timer runs as configured and notifies the user at the end of each session.*

Use Case

UC Name	***ThinkLink -UC-11***
Summary	*(Task Management and Progress Tracking): This requirement depicts the process of creating, managing, and tracking tasks during Pomodoro sessions.*
Dependency	*No dependencies*
Actors	***User (Student or Academic Staff), System***
Preconditions	*The user must be logged into the system and have access to the task management feature.*
Description of the Main Sequence	• ***Step 1:*** *User creates a list of tasks to complete during sessions.* • ***Step 2:*** *User edits or deletes tasks as needed.* • ***Step 3:*** *System tracks completed tasks* • ***Step 4:*** *System displays progress on the dashboard.* • ***Step 5:*** *Notifications remind users to start new sessions if inactive*
Description of the Alternative Sequence	• ***Step 2:*** *If the user attempts to edit a non-existent task, the system notifies them of the error.*
Non functional requirements	*Task creation, editing, and deletion should be performed within 2 seconds. The system should track up to 100 tasks simultaneously.*
Postconditions	*The list of tasks is updated and progress is accurately tracked.*

Use Case

UC Name	**ThinkLink-UC-12**
Summary	*(Flashcard Creation & Management): This use case describes the process of creating, organizing, and managing flashcards with text, images, or mathematical formulas.*
Dependency	*No dependencies*
Actors	***Content Provider***
Preconditions	*The content provider must be logged into the system.*
Description of the Main Sequence	• ***Step 1:*** *The content provider clicks on "Create Flashcard" button.* • ***Step 2:*** *The content provider creates flashcards by adding text, images, mathematical formulas and category.* • ***Step 3:*** *The content provider creates the submit button.* • ***Step 4:*** *The system saves the flashcards shows it in the relevant category.* • ***Step 5:*** *The content provider can edit, update, or delete flashcards as needed.*
Description of the Alternative Sequence	***Step 2:*** *If invalid content is added (e.g., unsupported file type, inappropriate material), the system notifies the content provider and prevents saving.*
Non-functional Requirements	*The system should support various formats (text, images, formulas) and allow smooth editing and categorization.*
Postconditions	*Flashcards are successfully created, organized into decks, and available for use.*

Use Case

UC Name	***ThinkLink-UC-13***
Summary	*(Flashcard Access & Study Modes) This use case describes how students can access flashcards and choose different study modes for effective learning.*
Dependency	*No dependencies*
Actors	***Student***
Preconditions	*Flashcard decks must be available in the system. The student must have access to the flashcard feature.*

Description of the Main Sequence	• ***Step 1:*** *The student browses or searches for flashcard decks using keywords, tags, or subjects.*
	• ***Step 2:*** *The system displays available flashcards based on the search.*
	• ***Step 3:*** *The student selects a study mode:*
	  ○ ***Standard Mode*** *– Flip front and back of cards.*
	  ○ ***Quiz Mode*** *– Multiple-choice or fill-in-the-blank.*
	  ○ ***Spaced Repetition Mode*** *– Adaptive review based on performance.*
	• ***Step 4:*** *The student can shuffle cards for randomized practice.*

Description of the Alternative Sequence	• ***Step 1****: If no matching flashcards are found, the system suggests related decks based on the student's search history or preferences.* • ***Step 4:*** *If the student wants a different study experience, they can switch between study modes at any time.*
Non-functional Requirements	*The system should provide fast search functionality and support seamless transitions between study modes.*
Postconditions	*The student successfully accesses and studies flashcards in the chosen mode.*

Use Case

UC Name	***ThinkLink-UC-14***
Summary	*(Search, Tracking & Gamification): This use case describes how students can search for flashcards, track their progress, and engage in gamified learning with points, badges, and challenges.*
Dependency	*No dependencies*
Actors	***Student***
Preconditions	*The student must have an active account and previously studied flashcards for spaced repetition tracking.*
Description of the Main Sequence	• ***Step 1:*** *The student searches for flashcards using keywords, tags, or subjects.* • ***Step 2:*** *The system uses the Spaced Repetition algorithm to suggest flashcards that need revision based on past performance.* • ***Step 3:*** *The system tracks progress.* • ***Step 4:*** *The system provide mastery analytics and revision suggestions.* • ***Step 5:*** *Students earn points and badges for completing revision tasks.* • ***Step 6:*** *Students can participate in weekly/monthly challenges where they compete on quiz scores or streaks to earn additional rewards.*
Description of the Alternative Sequence	*N/A (The process is continuous and allows updates anytime.)*
Non-functional Requirements	*The system should provide fast and accurate search results, real-time tracking, and smooth gamification elements without performance issues.*
Postconditions	*The student successfully revises flashcards, tracks their progress, and engages in challenges to enhance learning motivation.*

Use Case

UC Name	***ThinkLink-UC-15***
Summary	*(Job and Internship Listings)This use case describes how company representatives post job and internship listings and how students search for relevant opportunities.*
Dependency	*No dependencies*
Actors	*Company Representative,* ***Student***
Preconditions	*The company representative must have an authorized account to post listings.*
Description of the Main Sequence	• ***Step 1:*** *A company representative click on "Create Post" button.* • ***Step 2:*** *A company representative posts job and internship listings on the platform, providing details like company name, role description, and application deadlines.* • ***Step 3:*** *The company representative categorizes listings by industry, location, and job type (full-time, part-time, remote).* • ***Step 4:*** *Students search using filters (industry, location, job type)* • ***Step 5:*** *The system displays filtered results.*
Description of the Alternative Sequence	*N/A (The process is continuous and allows updates anytime.)*
Non-functional Requirements	*The system should support an intuitive search and filtering mechanism, ensuring quick and accurate results. It must handle a large number of listings efficiently.*
Postconditions	*Job and internship listings are successfully posted, categorized, and searchable by students.*

Use Case

UC Name	***ThinkLink -UC-16***
Summary	*(Profile Creation & Customization): This requirement depicts the creation and customization of a student's career profile in the system.*
Dependency	*No dependencies*
Actors	***Student**, Company Representatives, System*
Preconditions	*The student must have a valid university account and access to the system.*
Description of the Main Sequence	• ***Step 1:** Student click on "Create Profile" button.* • ***Step 2:** Student creates the career profile by entering details about their education, skills, and work experience.* • ***Step 3:** Student clicks on submit button.* • ***Step 4:** The student can edit their profile at any time to update relevant information.* • ***Step 5:** The company representative search the name of the student.* • ***Step 6:** The company representatives double click on the student profile and see their data.* • ***Step 7:** The student receive notification for each profile view and search.*
Description of the Alternative Sequence	*N/A (The process is continuous and allows updates anytime.)*
Non functional requirements	*-The profile creation process should not exceed 5 minutes.* *-The system must allow seamless updates with real-time data validation and user-friendly input forms.*
Postconditions	*A student's career profile is successfully created and stored in the system, with the option to update it as needed.*

Use Case

UC Name	***ThinkLink-UC-17***
Summary	*(Search & Application Process)This use case describes how students search for job or internship opportunities and apply, while companies manage applications and update statuses.*
Dependency	*No dependencies*
Actors	***Student**, Company Representative*
Preconditions	*Job and internship listings must be available in the system. The student must have an account to apply.*
Description of the Main Sequence	• ***Step 1:** A student searches for internships or job opportunities using keywords, location, job type, or experience level.* • ***Step 2:** The student selects a job or internship by double clicking on the job or internship post.* • ***Step 3:** Student applies directly by clicking on 'Apply' button* • ***Step 4:** Student fulfill necessary information and attach relevant documents (e.g., resume, cover letter).* • ***Step 5:** Student click on 'Submit' button.* • ***Step 6:** The company representatives shortlist candidates**,** contact them for interviews, or reject applications.* • ***Step 7:** The system updates the application status (e.g., shortlisted, interview scheduled, rejected)* • ***Step 8:** The system notifies students accordingly.*
Description of the Alternative Sequence	*N/A (The process is continuous and allows updates anytime.)*
Non-functional Requirements	*The system should provide a fast and accurate search experience, support secure document uploads, and deliver real-time status updates and notifications.*
Postconditions	*The student's application is submitted successfully, and the company representative can manage the application process effectively.*

Use Case

UC Name	***ThinkLink -UC-18***
Summary	*(Virtual Career Fairs & Events): This requirement describes the process of organizing and participating in virtual career fairs and webinars, allowing students to explore job opportunities and interact with company representatives.*
Dependency	*No dependencies*
Actors	***Student**, Company Representative*
Preconditions	*The students must have been signed in the system to have access.*
Description of the Main Sequence	• ***Step 1:** Company representative creates a virtual career fair or webinar with event details.* • ***Step 2:** Company representatives publishes the event by clicking on "Public" button.* • ***Step 3:** The system sends a notification to students.* • ***Step 4:** Students register for the event.* • ***Step 5:** Students join the virtual event at the scheduled time.* • ***Step 6:** Company representative presents job opportunities and career insights.* • ***Step 7:** Students participate in Q&A sessions and network with company representatives.*
Description of the Alternative Sequence	• ***Step 3**: If the student does not register on the virtual career fair he will get notified again one day before the virtual career fair or webinar.*
Non functional requirements	*The system must support at least **1,000 simultaneous attendees** in a single virtual career fair.* *Event notifications must be sent **at least 24 hours and 1 hour** before the event starts.* *The platform should ensure **high-quality streaming** and **interactive features** (live Q&A, polls, and chat).*
Postconditions	*Students gain insights into job opportunities and career options.* *Company representatives can connect with potential candidates and receive student inquiries.*

Use Case

UC Name	***ThinkLink -UC-19***
Summary	*(Past Papers Process): This requirement depicts how students interact with the system to upload, search, and manage past papers, while administrators moderate the content.*
Dependency	*No dependencies*
Actors	*Student, Administrator, System*
Preconditions	*The student must be logged into the system with a valid university account.*
Description of the Main Sequence	• ***Step 1:*** *Student clicks on the "Upload Past Paper" on the left of screen.* • ***Step 2:*** *Student uploads past exam papers, quizzes, and assignments, categorizing them by university, course, subject, and academic year.* • ***Step 3:*** *Student filters past papers by university, courses, subject, and academic year.* • ***Step 4:*** *Student preview past papers by double clicking on the the past paper.* • ***Step 5:*** *Student downloads the past papers if wanted.* • ***Step 6:*** *Student rates, reviews, and provides feedback on past papers to ensure quality and relevance.* • ***Step 7:*** *Administrator uses moderation tools to review and remove content that violates guidelines or is of low quality.*
Description of the Alternative Sequence	• ***Step 7****: If flagged content is found, the administrator removes or requests edits for the uploaded past papers.*
Non functional requirements	*-The system should handle up to 500 simultaneous uploads and 1000 searches per hour.* *-The upload process should not exceed 3 minutes and must support multiple file formats (PDF, DOCX).*
Postconditions	*The past papers are available in the system's database, categorized and searchable, with moderation to ensure quality.*

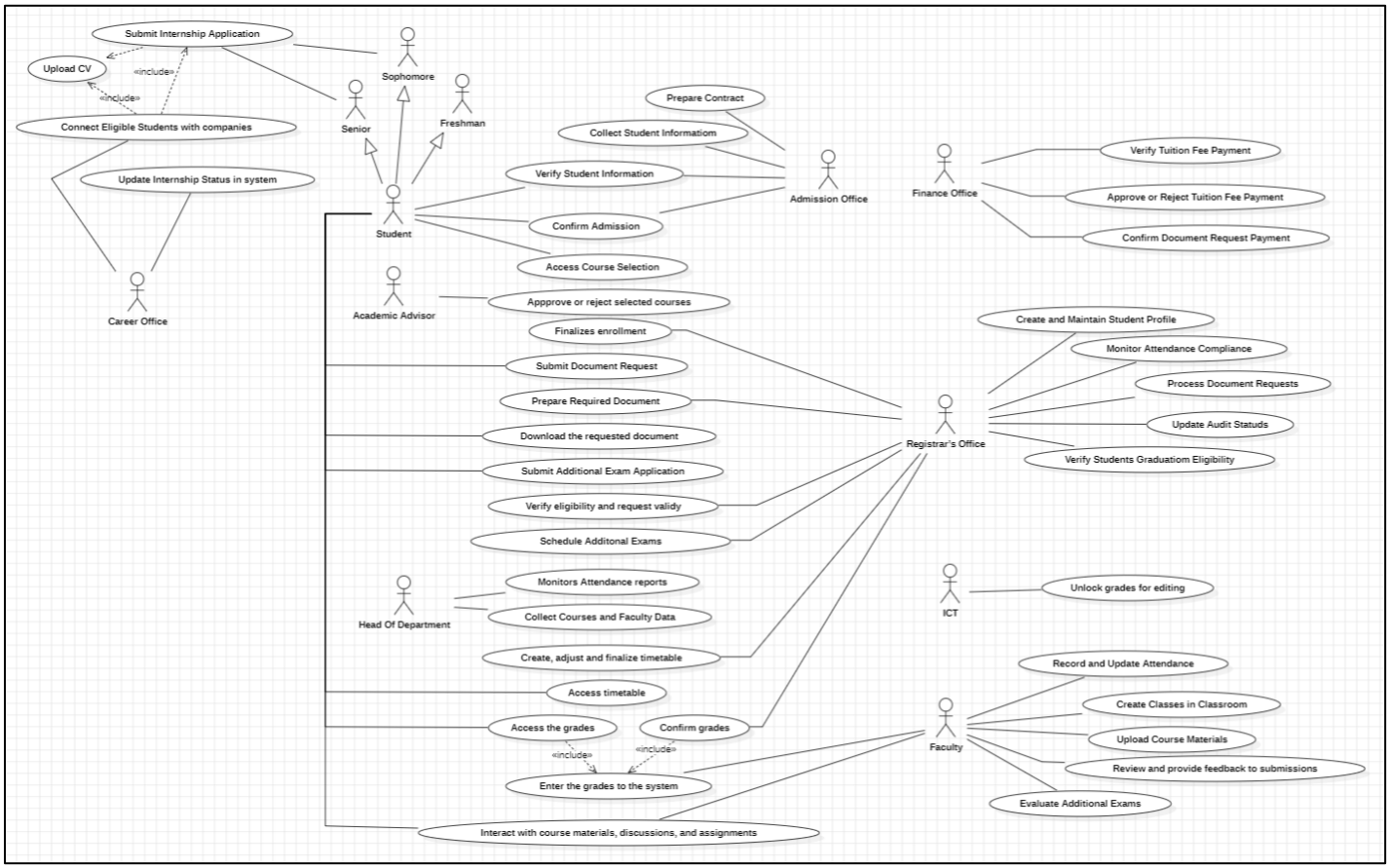
Use Case

UC Name	***ThinkLink -UC-20***
Summary	*(Textbooks): This requirement depicts a digital library system where students can browse, download, and customize textbooks, while content providers upload and manage materials, and regulatory bodies ensure compliance with academic standards.*
Dependency	*ThinkLink-UC-1*
Actors	***User**, Content Providers, Regulatory Bodies*
Preconditions	*The user is logged into the system and has access to the digital library where textbooks are categorized by subject, course, and academic year.*
Description of the Main Sequence	• ***Step 1:** The user selects a category to search for textbooks.* • ***Step 2:** The system displays relevant textbooks, and the user selects one for more details.* • ***Step 3:** The user downloads the textbook in their preferred format and opens it in the integrated reader.* • ***Step 4:** The user customizes the reading mode or language and interacts with the textbook. (highlight text, add annotations, create bookmarks for easy reference)* • ***Step 5:** The user can browse and download additional textbooks as needed.*
Description of the Alternative Sequence	• ***Step 3:** If no textbooks are found in the selected category, the system informs the user that no textbooks are available. The user is then given options to refine their search.*
Non functional requirements	*The system must support up to 10,000 active users at any given time without delays, crashes, or degraded performance.*
Postconditions	*The user has successfully downloaded or accessed the selected textbook, customized the reading experience (including language, format, and reading mode), and is able to interact with the textbook through highlighting, annotations, and bookmarks.*

Use Case

UC Name	***ThinkLink -UC-21***
Summary	*(Seminars Process): This requirement depicts the creation, management, participation, and recording of seminars organized by students and institutions.*
Dependency	*No dependencies*
Actors	***Student**, Content Providers, System*
Preconditions	*Students and institutions must have valid accounts and appropriate permissions to create or register for seminars.*
Description of the Main Sequence	• ***Step 1:** Content Providers create and manage seminars, entering details such as topic, speaker(s), date, time, and delivery mode (online or in-person).* • ***Step 2:** Students register for seminars* • ***Step 3:** Student receive automated reminders via email or in-app notifications.* • ***Step 4**: Student engages with features such as live chat, Q&A sessions, and polls to enhance engagement.* • ***Step 5:** The system stores recorded seminars.* • ***Step 6:** Student click on 'Past Seminars' menu.* • ***Step 7:** Student double click on the seminar post and have access on the seminar materials.*
Description of the Alternative Sequence	• ***Step 4:** If interactive features are disabled due to technical issues, seminars proceed in lecture-only mode, and the session is still recorded.*
Non functional requirements	*-The system should support at least 300 concurrent seminar participants.* *-Reminders must be sent at least 24 hours and 1 hour before the seminar.* *-Recordings must be available within 1 hour after the seminar ends.*
Postconditions	*The seminar is successfully created and managed, students are notified, and the seminar recording is stored and available for future access.*

UIS – Use Case diagram



ThinkLink – Use Case diagram